CSE 220 – Programming in C

Final Exam

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Specify if each of the following statements is True or False (14 pts).**

|  |  |
| --- | --- |
| The memory for the array tmp declared as: int tmp[20];  can be deallocated using: free(tmp); |  |
| The size of a union is the size of the longest element in the union |  |
| Header files are compiled when the main program that includes them is compiled |  |
| Static variables declared inside a function are not destroyed on exit from the function call |  |
| The call strcmp(“Good Morning”, “Bonjour”) returns 0 |  |
| The statement: long \*ptr = (long\*) malloc(100)  allocates enough memory to store 100 long integers |  |
| Any struct variable may be assigned to another struct variable as long as both structs have the same member variables (matching both type and name) |  |

1. **What is the output of the following programs:**

**Program A: (2 pts)**

#include<stdio.h>

int main() {

int i=1, j=2, k=3, m=9.4;

j \*= i++\*j-(--k)+ ++m/j;

printf(“i = %d, j = %d, k = %d, m = %d”, i, j, k, m);

return 0;

}

**Program B: (2 pts)**

#include<stdio.h>

#include<string.h>

int main() {

char \*str = “I like roses”;

char str2[100] = “Reds are my favorites”;

printf(“%s”, strcpy(str2, strcat(str2, str)));

return 0;

}

**Program C: (2 pts)**

#include<stdio.h>

struct GameResult {

int score;

char player[25];

};

int main() {

struct GameResult logs[] = {{102, "Mark"},

{220, "The King"},

{24, "Newbie"},

{132, "Mark"},

{36, "Newbie"}};

printf("%d ", logs[1].score);

struct GameResult \*ptr = logs+2;

if ( (\*(logs+2)).score > 100){

printf("%s\n", (\*(ptr+2)).player);

} else {

printf("%s\n", (--ptr)->player);

}

return 0;

}

1. **Write a program that reads an integer n from the user and outputs the following pattern: (10 pts)**

1

1 2

1 2 3

1 2 3 4

…

…

1 2 3 4 … n

1. **Write a function ‘trim’ that takes a string and removes all leading and trailing spaces from it. The result is returned as a new string. (10 pts)**
2. **Write a recursive function that given an array of char of size n, returns 1 if all elements in it are uppercase alphabet letters. (10 pts)**
3. **The following program reads the name and grade of 10 students from standard input into an array and saves to a file the records with a passing grade. The file name is read from command line. The program contains some errors. Fix five of them. (10pts).**

#include “stdio.h”

int main(int argc, char \*argv[]) {

struct StudentRecord {

char \*name;

int grade;

};

StudentRecord recs[10];

File \*fout = fopen(“passingStudents.txt”, “r”);

for (int i=0; i<=9; i++) {

scanf(“%s %d”, recs[i].name, recs[i].grade);

if (recs[i].grade >= 60) {

fprintf(fout, “%s %d”, recs[i].name, recs[i].grade);

}

}

close(fout);

return 0;

}

1. **The following program reads a sentence from standard input. The program should ask the user to enter two numbers corresponding the space numbers in the string (for example, the user enters 3 and 7 to denote the 3rd and 7th space in the string. The program then locates these spaces and constructs a string equal to the substring between the spaces located, not including the spaces. Finish the program: (20 pts)**

/\* include any libraries needed (3 pts) \*/

int main(){

/\* Declare a string as a pointer to char and allocate memory for it to hold 1000 chars (2 pts) \*/

/\* Ask the user to enter a string. Read it from standard input. Make sure you do not read more than the string can hold (3 pts)\*/

/\* Ask the user to enter the indices for the spaces \*/

int i, j;

printf(“Enter two integers denoting the space numbers:\n”);

scanf(“%d %d”, &i, &j);

/\* Check that both numbers are less than the length of the string read. Display error message and exit if they are not (2 pts) \*/

/\* Declare two pointers. Make the first one point to the ith space and the second one point to the jth space (5 pts) \*/

/\* Construct a new string as a substring of the original one, between (exluding) the ith and jth spaces and print it to the standard output (5 pts) \*/

return 0;

}

1. **What is the output of the following program? (10pts)**

#include <stdio.h>

int a = 10;

int c[3] = {1, 2, 3};

int f(int a){

a++;

printf(“F. %d\n”, a);

return a;

}

void g(int array[], int n) {

for (int idx = 0; idx<n; idx++)

array[idx] = 2\*array[idx];

}

int h() {

a = a/2;

printf(“H. a = %d\n”, a);

return a;

}

int main() {

int u = 20;

u = h();

printf(“M1. u = %d\n”, u);

int b = 10;

f(b);

printf(“M2. b = %d, a = %d \n”, b, a);

g(c, 2);

printf(“M3. C0 = %d, C2 = %d \n”, c[0], c[2]);

return 0;

}

1. **Write a program that reads from the file essay.txt and copies all words that start with the letter t into the file output-t.txt, writing each 3 words on a separate line. (15 pts)**